Speech Intelligibility Prediction Toolbox

An easy way for modeling intelligibility and quality of speech

October 8, 2010

User Guide

Fraunhofer Institute for Digital Media Technology IDMT
Project Group Hearing, Speech and Audio Technology
Oldenburg

Telephone: +49 441 2172-433
Email: sip-toolbox@idmt.fraunhofer.de

Contents

1 Introduction **3**
- 1.1 About SIP-Toolbox . . . 3
- 1.2 Requirements . . . 3
- 1.3 Demo version . . . 3
- 1.4 Installation . . . 3
  - 1.4.1 Install PEMO-Q copy protection . . . 4
  - 1.4.2 Install SIP-Toolbox . . . 4

2 Getting started **5**

3 Signal selection **6**

4 Signal generation to evaluate signal processing algorithms (Hagerman & Olofsson) **7**
- 4.1 Batch processing using the method of Hagerman & Olofsson . . . 9

5 Module selection **11**
- 5.1 Module Speech Intelligibility . . . 11
  - 5.1.1 Calculation of speech intelligibility . . . 13
- 5.2 Module Loudness . . . 13
  - 5.2.1 Loudness prediction . . . 15
- 5.3 Module Speech Quality . . . 16
  - 5.3.1 Select the reference signal . . . 17
  - 5.3.2 Assessment of speech quality . . . 17
- 5.4 Module Impulse Respone Analysis . . . 18
  - 5.4.1 Perform room impulse response analysis . . . 20

6 Main menu bar **21**
- 6.1 Menu item Setup . . . 21
  - 6.1.1 Load . . . 21
  - 6.1.2 Save . . . 21
  - 6.1.3 Batch Processing . . . 21
  - 6.1.4 Hagermann & Olofsson . . . 24
  - 6.1.5 Preferences . . . 24
  - 6.1.6 Reset . . . 24
- 6.2 Menu item Audiogram . . . 24
- 6.3 Menu item Binaural Options . . . 25

1 Introduction

1.1 About SIP-Toolbox

Assessment of speech in all situations

The "Speech Intelligibility Prediction Toolbox" (SIP-Toolbox) developed in the project group Hearing, Speech and Audio Technology at the Fraunhofer IDMT offers a quick and easy prediction of the main factors affecting speech quality for different situations. It can be used to easily compare different models and thereby assist the user to select the most suitable models for specific applications. The SIP-Toolbox designed in MATLAB® offers versatile utilities to import, process and represent data. For users with no access to MATLAB®, we also offer a stand-alone version of the SIP-Toolbox.

1.2 Requirements

The software SIP-Toolbox is delivered as an executable file (exe) for MICROSOFT WINDOWS 32bit operation systems or as a "p-coded" function of MATLAB® version 2008b or higher for MICROSOFT WINDOWS 32bit operation systems. The SIP-Toolbox contains a copy protection in terms of one or two USB dongles (depending on the features). We recommend a minimum screen resolution of 1280 x 1024 pixels.

1.3 Demo version

A demo version of the SIP-Toolbox is available as a "p-coded" function for MATLAB® version 2008b and higher. Additionally, a stand-alone version of the SIP-Toolbox without the use of MATLAB® is available.

The demo version contains all modules and functions of the full version, except for the following limitations

- No processing and assessment of user-selected signals
- Saving and loading of results are disabled
- Limited selection of predefined audio signals
- Batch processing is disabled
- "PEMO-Q" speech quality model (see 5.3) is disabled
- No hearing impairment can be included in the models

If you have any questions about the demo version of the SIP-Toolbox or if you are interested in further information, please do not hesitate to contact us.

1.4 Installation

the installation may depend on the features of your SIP-Toolbox. If you are using a demo version or a version without the speech quality model "PEMO-Q", please continue with section 1.4.2.

If your version contains "PEMO-Q", you have to install the copy protection of this module first as described in section 1.4.1. PLEASE DO NOT insert the copy protection before you have installed the driver.

1.4.1 Install PEMO-Q copy protection

To install the PEMO-Q copy protection driver please unzip the installation files in any folder. Then open a windows command line (WINDOWS START → RUN → "cmd"), go to the subdirectory "Hinstall" in the installation folder and type

```
hinstall -i
```

in the command line and press ENTER. The installation will take some minutes. After finishing the installation, insert the USB dongle. MS WINDOWS will now integrate the copy protection. To install the SIP-Toolbox software continue with the next step.

1.4.2 Install SIP-Toolbox

Stand-alone version (without the use of MATLAB®)

The installation package is delivered as a zip file with the program and installation files. Unzip the files into any folder. The structure of the unpacked folders is fixed. PLEASE DO NOT MOVE OR RENAME any folder or file. The unpacked zip file contains a folder named "Installation".

To work with the stand-alone version of the SIP-Toolbox you need to install the MATLAB® runtime library. Therefore please execute the file

```
MCRInstaller.exe
```

in the "Installation" folder. You can choose any local directory for the MATLAB® runtime library.

After finishing the installation process, please insert your USB dongle. You can now start the SIP-Toolbox by executing the file `SIP_Toolbox(_demo).exe`.

P-coded function for MATLAB® 2008b or higher

The installation packages is delivered as a zip file including the program files. Unzip the files into any folder. The structure of the unpacked folders are fixed. PLEASE DO NOT MOVE OR RENAME any folder or file. To work with the SIP-Toolbox please start MATLAB® , go to the folder where you unzipped the files, insert the USB dongle and start the file `SIP_Toolbox(_demo).p` in MATLAB®.

2 Getting started

The SIP-Toolbox is a modularly structured software which offers an easy handling via a graphical user interface. After starting the software the main window is opened as shown in Figure 1. This main window is organized in two main sections: the signal selection and parameter setting on the left and the module and model selection on the right side.

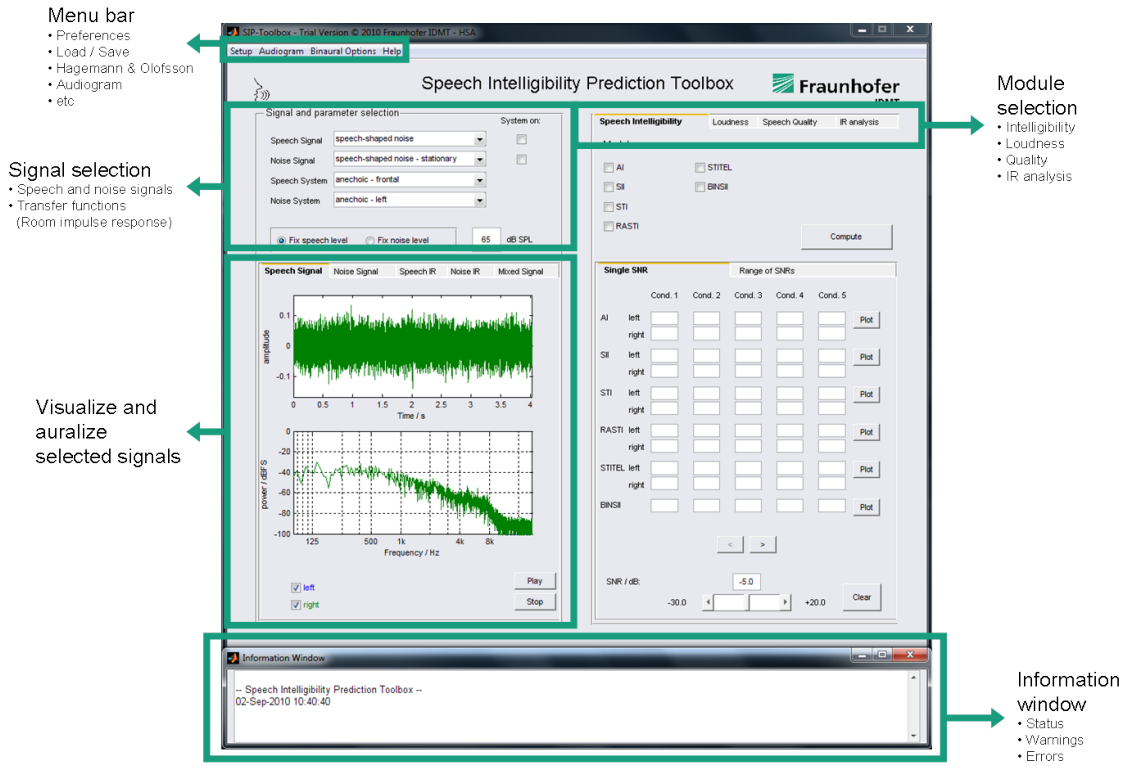

Figure 1: Overview of the SIP-Toolbox.

The menu bar offers a quick access to global functions like preferences and the load/save menu. The menu item “Batch Processing” allows an automatic processing of more than one signal. Further information about the batch processing in the SIP-Toolbox is given in section 6.1.3. All menu items are explained in detail in section 6.

You will find practical information about the item “Hagerman & Olofsson” from the menu bar in section 4. This menu item allows you to generate special signals, which can be used e.g. to evaluate noise reduction algorithms. It is an implementation of the method described by Hagerman & Olofsson [1].

The left side of the main window is used to select the signals you want to work with and to set the acoustic parameters. The signals are automatically visualized in the time and frequency domain. On the right, side you can select a module to evaluate the selected signals. The modules are described separately in section 5.
The information window at the bottom shows warnings and information of processed calculations of the SIP-Toolbox. This protocol can be saved using the menu bar.

3 Signal selection

The signal selection (see Figure 2) allows you to load PCM audio files (*.wav) into the SIP-Toolbox. The pop-up menus “Speech Signal” and “Noise Signal” provide a comfortable way to select predefined signals or to integrate your own signals. Furthermore, it is possible to add a transfer function in form of a room impulse response (RIR) to the selected signals. You add the transfer function to the signal (convolution) by activating the check box below “System on”. Room impulse respones are supported as wave file (*.wav), as text file (*.txt) and as a MATLAB® file (*.mat). The internal sampling frequency of the SIP-Toolbox is 44.1 kHz. Signals and transfer functions differing from 44.1 kHz sampling rate are automatically resampled. Single-channel and dual-channel signals are supported. If you use single-channel signals, the channel will be dublicated to generate the second channel.

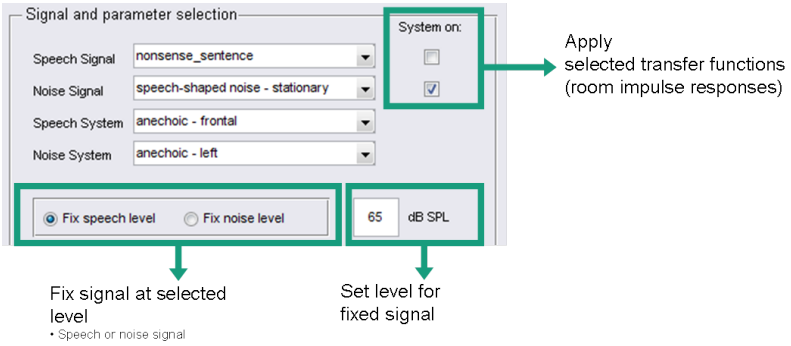

Figure 2: Zoom of signal selection in the SIP-Toolbox.

The calibration of the signals is based on the long-term RMS value. The SIP-Toolbox calibrates the digital signals to sound pressure level in dB SPL using a reference pressure at 20^{-6} pascal. To process speech and noise signals at a specified signal-to-noise ratio, you have to define which signal (speech or noise) has a fixed level. The level of the other signal is then adjusted according to the desired signal-to-noise ratio (see Figures 2 and 3).
Below the signal selection part, signals and transfer functions are visualized in the time and frequency domain. Figure 3 illustrates this for the mixed signal consisting of speech and noise. The panels provide a comfortable way to switch between the visualization of the different signals (speech, noise, mixed signal or transfer functions). As shown in Figure 3 for the “mixed signal” you are able to set the signal-to-noise ratio in dB using the slider. Additionally, you can define which channel of the signal should be displayed.

This part of the SIP-Toolbox also allows you to listen to the signals. For playback the signals will be amplified (or attenuated) to a full scale RMS value of -25 dB to ensure distortion free listening. This adjusting is independent of the visualization.

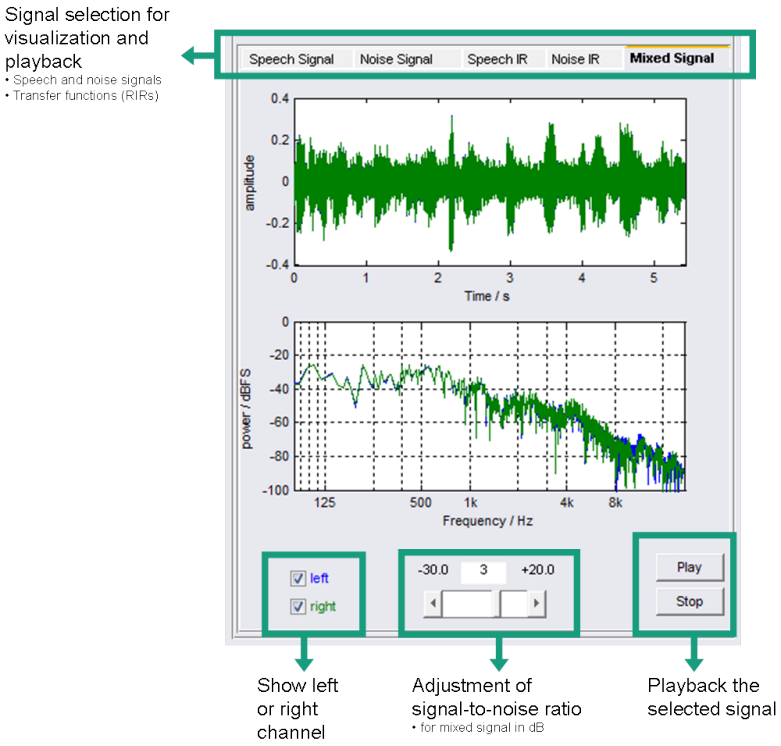

Figure 3: Visualization of the mixed signal (speech and noise) in the time and frequency domain.

4 Signal generation to evaluate signal processing algorithms (Hagerman & Olofsson)

This section describes the possibility to evaluate the effect of signal processing algorithms, for instance algorithms used in hearing aids. It is based on research results of Hagermann & Olofsson [1].
Firstly, two mixed signals consisting of speech and noise are generated according to

$$a_{in}(t) = s(t) + n(t)\,, \tag{1}$$
$$b_{in}(t) = s(t) - n(t) \tag{2}$$

The difference between the two signals consists only in the opposite sign of the noise signal. Both signals $a_{in}(t)$ and $b_{in}(t)$ are now ready to be processed by an algorithm. The results of the process are recorded and can be represented in simplified way by[1]

$$a_{out}(t) = s'(t) + n'(t)\,, \tag{3}$$
$$b_{out}(t) = s'(t) - n'(t)\,. \tag{4}$$

From these two processed signals $a_{out}(t)$ and $b_{out}(t)$, the speech signal $s'(t)$ and the noise signal $n'(t)$ are extracted separatly. Now it is possible to compare the processed speech signal $s'(t)$ with the original speech signal $s(t)$ to evaluate the effects of the algorithm. Analogously, the method also works for the noise signals.

[1]For restrictions and limitations of the method see [1].

Step 1: Select the signals and transfer functions you want to work with according to the SIP-Toolbox signal selection. If you want to work with the mixed signal please set the desired SNR ("Mixed Signal"). Now generate signals according to $a_{in}(t)$ and $b_{in}(t)$ from your selected signals using the main menu bar ("Setup" → "Hagemann & Olofsson" → "Generate test signals"), see Figure 4. You will receive a request if all settings and signals are correct. If you answer with "Yes" you have to choose a file name and a storage location. Only one file name (for example `testsig`) is necessary, the second file will automatically be generated. The files are saved with a special name extension, like

- `testsig_SplusN.wav` according to $a_{in}(t)$,
- `testsig_SminusN.wav` according to $b_{in}(t)$.

You will be informed about the successful completion in the "information window".

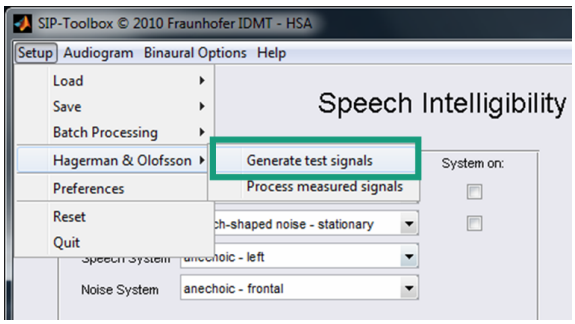

Figure 4: Menu item to generate signals according to $a_{in}(t)$ and $b_{in}(t)$.

Step 2: Both signals, `testsig_SplusN.wav` and `testsig_SminusN.wav`, are now available for the signal processing with an algorithm. It is important that both signals are processed and the two results $a_{out}(t)$ and $b_{out}(t)$ are recorded and saved separately to a file. In the next step these processed signals are loaded in the SIP-Toolbox to evaluate the effect of the algorithm.

Step 3: To add the processed signals to the software please use the function "Process measured signals" in the menu ("Setup" → "Hagemann & Olofsson" → "Process measured signals") shown in Figure 5. In the example, the processed signals are

`processed_testsig_SplusN.wav` according to $a_{out}(t)$,

`processed_testsig_SminusN.wav` according to $b_{out}(t)$.

There are to subsequent dialogs to select the processed file according to "Speech plus noise" and for the file according to "Speech minus noise". Finally, the software extracts the speech signal $s'(t)$ from both processed signals $a_{out}(t)$ and $b_{out}(t)$ (see [1]). Similarly, the noise signal $n'(t)$ is obtained. To save these files please select a storage location and a file name. Only one file name is necessary, the second file is automatically generated. By default the files are named like

`HagOlof_Processed_Signals_speech.wav` according to $s'(t)$,

`HagOlof_Processed_Signals_noise.wav` according to $n'(t)$.

After the saving is completed, these files are automatically inserted in the SIP-Toolbox as speech and the noise signal. Now you are able to evaluate the files and compare the results

with the assessment of the original files $s(t)$ and $n(t)$. You will find more precise information on all modules in the following sections.

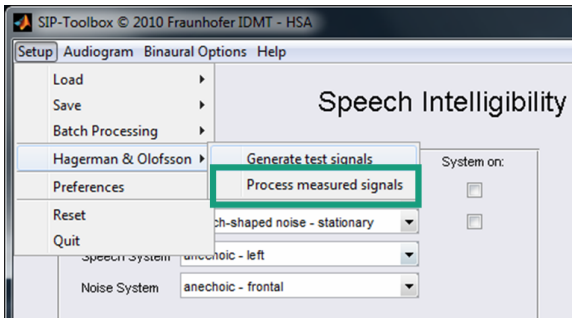

Figure 5: Menu item to process the measured signals according to $a_{out}(t)$ and $b_{out}(t)$.

4.1 Batch processing using the method of Hagerman & Olofsson

The SIP-Toolbox also provides the opportunity to extract the speech and the noise signals from a large collection of processed signals (batch processing) according to the scheme by HAGERMAN & OLOFSSON. Therefore you need the processed signals, speech plus noise (`..._SplusN.wav`) and speech minus noise (`..._SminusN.wav`). The functionality and the operating of this batch processing is described below.

The software package contains four different example signals to illustrate batch processing. These example signals are located in the folder `signals\processed_...`. They are composed of speech and white noise at different signal-to-noise ratios and have been processed by two noise cancellation algorithms.

folder `.\signals\processed_SminusN\` :

`01_proc_speech_whiteNoise_0dBSNR_SminusN_algo1.wav`

`02_proc_speech_whiteNoise_+5dBSNR_SminusN_algo2.wav`

folder `.\signals\processed_SplusN\` :

`01_proc_speech_whiteNoise_0dBSNR_SplusN_algo1.wav`

`02_proc_speech_whiteNoise_+5dBSNR_SplusN_algo2.wav`

These signals are already separated into `SplusN` and `SminusN` and placed in the respective folder. For the batch processing it is important that all signals are stored in the correct folder and named in a consistent order (e.g. by numbering). The batch processing to extract the speech and noise signal from processed signals can now be started in the menu item "Setup" → "Batch Processing" → "Hagerman & Olofsson Signal Generation". Before, it is necessary to create a text file with special parameters to control the batch processing. Figure 6 shows an example text file (`testHand0_SigGen_batch_in.txt`) to initialize the batch processing. The signal paths (for `SplusN` und `SminusN`) have to be written correctly in the text file. It is possible to set an absolute path or a relative path. To set a relative path please use following syntax (`.\...\`). In addition,

you have to name directories, where the extracted speech and noise signals will be stored. All steps will be explained more precisely in the following.

```
%%%%%%%%%%%%%%%%%%%%%%%%%
%  DIRECTORY SELECTION   %
%%%%%%%%%%%%%%%%%%%%%%%%%
% Measurement signals for speech+noise and speech-noise
SPEECH_PLUS_NOISE_SIGNAL_DIR:    .\signals\processed_SplusN
SPEECH_MINUS_NOISE_SIGNAL_DIR:   .\signals\processed_SminusN

% target directories to save resulting speech and noise signals
SPEECH_TARGET_DIR:               .\signals\Hand0_speech
NOISE_TARGET_DIR:                .\signals\Hand0_noise
```

Figure 6: Example text file (`testHand0_SigGen_batch_in.txt`) to control the batch processing according to Hagerman & Olofsson.

Step 1: Separate the processed signals according to `SplusN` and `SminusN` and place the files in the correct folders. Name the files consistently, for example by numbering them.

Step 2: Open or create text file, for instance `testHand0_SigGen_batch_in.txt`, and insert the paths of the source and result files. Please use the correct syntax according to the example text file.

Step 3: To start the batch processing select the item "Setup" → "Batch Processing" → "Hagerman & Olofsson Signal Generation" from the menu bar. Now there is a request to select the correct text file. Select the text file, for instance `testHand0_SigGen_batch_in.txt`, and the extraction for the speech and noise signal will start. You will be informed about the completion on the screen.

Now, the extracted speech and noise signals are located in the respective folders, for example

- `.\signals\Hand0_noise`
- `.\signals\Hand0_speech` .

These signals are now ready to be evaluated in the SIP-Toolbox. For convenience, you can assess these signals also using batch processing. The previously described signal generation according to Hagerman & Olofsson additionally creates a new text file, which contains ready set parameters to evaluate the speech intelligibility using batch processing. Figure 7 shows this text file for the processed example files. During the extraction the signal-to-noise ratio is determined and written to the text file. Further information about the batch processing for the quality and intelligibility assessment of speech is given in section 6.1.3.

```
% BATCH MODE HandO_SigGen
% Signals computed from: testHandO_SigGen_batch_in.txt
% Time stamp: 31-Mar-2010  16:43:39:

%%%%%%%%%%%%%%%%%%%%%%
%  SIGNAL SELECTION   %
%%%%%%%%%%%%%%%%%%%%%%
SPEECH_SIGNAL_DIR:  .\signals\HandO_speech
NOISE_SIGNAL_DIR:  .\signals\HandO_noise
SPEECH_RIR_DIR:
NOISE_RIR_DIR:

%%%%%%%%%%%%%%%%%%%%%%
% PARAMETER SELECTION %
%%%%%%%%%%%%%%%%%%%%%%
FIXSIGNAL:[S]
FIXLEVEL:[65]
SNR:[3.2368      5.4005]

%%%%%%%%%%%%%%%%%%%%%%
%  MODEL SELECTION    %
%%%%%%%%%%%%%%%%%%%%%%
AI
SII
STI
```

Figure 7: Batch text file (`HandO_Batch_processing_SpeechIntel_1.txt`) to evaluate the speech intelligibility with presettings created by the batch processing according to Hagerman & Olofsson.

5 Module selection

The SIP-Toolbox contains different modules. You can customize your SIP-Toolbox by selecting the modules needed for your application. Currently, the following modules are available:

- Speech Intelligibility
- Speech Quality
- Loudness
- Impulse Response Analysis

Each module contains several models and objective measures. These models offer the opportunity for comparative evaluation of the selected signals and communication situations.

5.1 Module Speech Intelligibility

The module "Speech Intelligibility" offers several objective measures to assess the intelligibility in a communication situation as shown in Figure 8. The models are described in the following.

AI: Calculation of the intelligibility according to the articulation index principle (ANSI S3.5-1969)[2]. Speech and noise signals are split up in 20 frequency bands (`Kryter`, JASA 1969). In every band the signal-to-noise ratio is evaluated and combined with a weighting function. The weighting function is independent of the speech material.

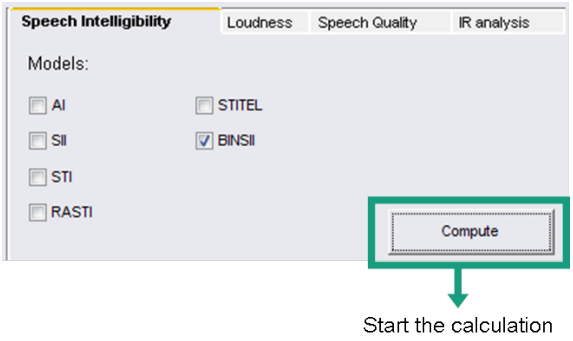

Figure 8: Zoom on the implemented measures within the speech intelligibility module.

SII: Calculation of the intelligibility according to the speech intelligibility index principle (ANSI S3.5-1997)[3]. The SII is an updated version of the articulation index, splitting up speech and noise signal into 21 critical bands (frequency bands by `Zwicker` [16]) and weighting the signal-to-noise ratio in each band. The weighting function depends on the used speech material. In this implementation the weighting function *SPIN* for speech in noise is used. Roughly, SII values above 0.75 indicate good speech intelligibility and SII values below 0.45 are related to poor intelligibility.

BINSII: Binaural extension of the SII according to Beutelmann & Brand [5]. It uses both ear signals to predict the intelligibility of speech. This model is able to predict a binaural benefit in conditions with spatially separated speech and noise sources.

STI: Calculation of the intelligibility according to the speech transmission index principle (IEC 60268-16)[4] using 7 octave bands and a weighted rating of the signal-to-noise ratio in each band. Additionally, the STI accounts for time-domain distortions, for example reverberation, by calculating the modulation transfer function. Here the modulation transmission in each octave band is determined for 14 different modulation frequencies. In the SIP-Toolbox the modulation transfer function is determined from the selected room impulse response (RIR) (`Schröder`, 1981). The classification of the STI is shown in Figure 9.

Figure 9: STI classification according to IEC 60268-16.

STITEL: Simplified version of the STI (IEC 60268-16) using 7 octave bands but only one modulation frequency in each octave band. TEL in STITEL stands for telecommunication systems.

RASTI: Simplified version of the STI (IEC 60268-16) using only two octave bands and four or five modulation frequency in each octave band, respectively. RA in RASTI stands for Rapid or Room Acoustics.

5.1.1 Calculation of speech intelligibility

There are two different ways to calculate intelligibility of speech using the SIP-Toolbox. You can assess the communication situation for one adjustable SNR (Figure 10) or for an adjustable range of SNR values (Figure 12).

Prediction of the speech intelligibility for a single SNR value

First you have to set an SNR value in dB using the slider (Figure 10). To start the calculation using the selected models press the button “Compute”.

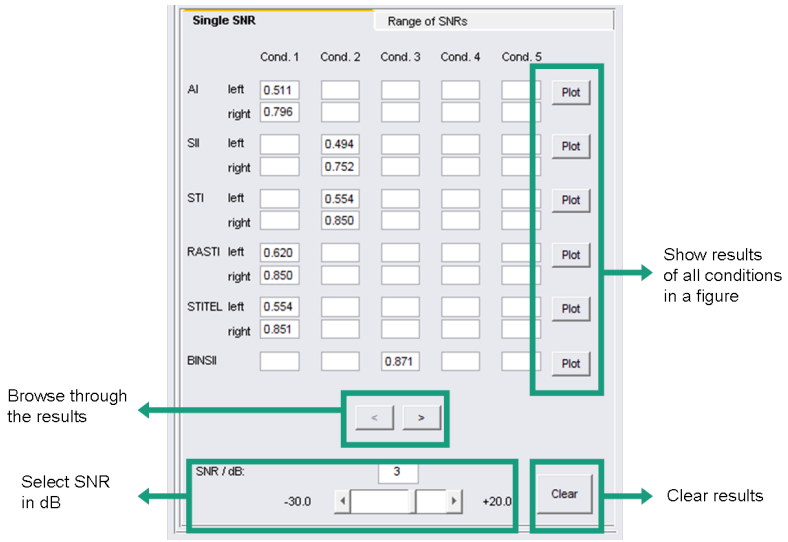

Figure 10: Calculation of the speech intelligibility for an adjustable signal to noise ratio (SNR) in dB. It is possible to assess more than five conditions without clearing the results.

For the monaural models, speech intelligibility is calculated and displayed separately for each channel. Every calculation is set as a new condition and the results are displayed in the corresponding column. At maximum it is possibly to show 50 result values before it is necessary to save and/or clear the result display. For a better presentation and to easily compare the results the SIP-Toolbox provides the possibility to plot all processed calculations in a separate figure by using the button “Plot”. Furthermore, it is possible to display all parameters from previous calculations, like signal name or SNR value by moving the mouse cursor over the text “cond” above the displayed results, see Figure 11.

Prediction of the speech intelligibility for a range of SNR values

To assess the speech intelligibility for a range of SNR values you can select the range and the calculation steps in dB, see bottom of Figure 12. The calculations will start after pressing the “compute” button and the results are plotted as a function of SNR. Within the result figure it is possible to mark values using the mouse cursor. You can also export the figure for later use.

5.2 Module Loudness

The module “Loudness” contains several models and objective measures as shown in Figure 13. These models offer the opportunity to comparatively evaluate the selected signals and communi-

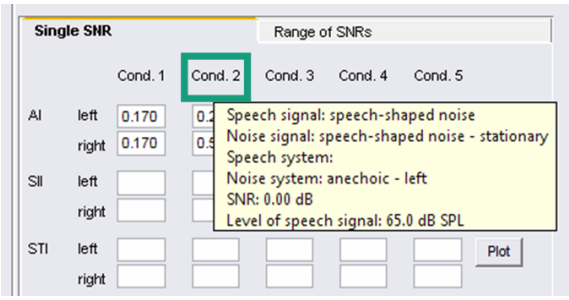

Figure 11: Display all set parameters used during the calculation by moving the mouse cursor over the corresponding word "cond".

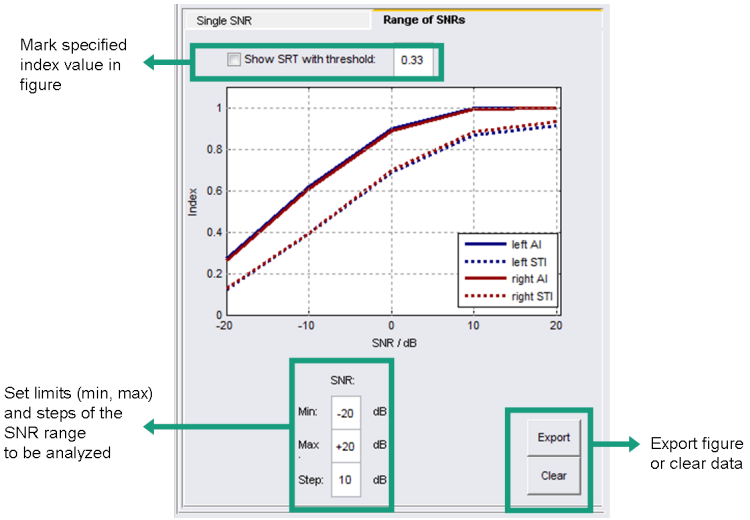

Figure 12: Calculation of speech intelligibility for a range of signal-to-noise ratios. The range and calculation steps are adjustable.

cation situations. The implemented loudness models are described in the following.

DIN 45631 / A1: Calculation of loudness level and loudness from the sound spectrum - Zwicker method - Amendment 1: Calculation of the loudness of time-variant sound [6].

DIN 45631 / ISO 532B: Calculation of loudness level and loudness from the sound spectrum - Zwicker method. This method is only appropriate for time-stationary sounds [7].

Glasberg & Moore 2002: Time varying loudness (TVL) [8]. The method is appropriate for time-variant sounds. The signals are resampled within the algorithm to 32 kHz.

Glasberg & Moore 1997: A much older but well-known loudness model of `Glasberg & Moore` [9] to predict the loudness sensation of time-stationary sounds.

All loudness models are calibrated such that a two channel sinusoidal signal of 1 kHz at 40 dB SPL results in loudness values between $0.98 \leq N \leq 1.01$ Sone, depending on the loudness model.

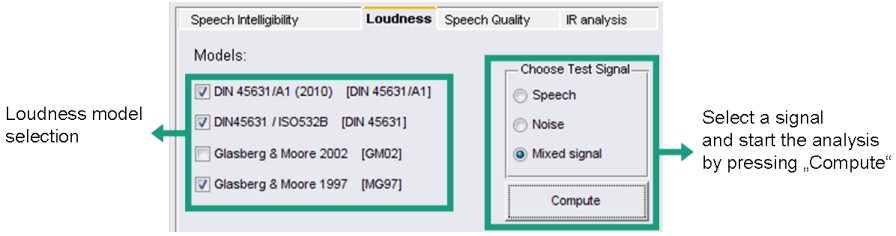

Figure 13: Zoom of the SIP-Toolbox loudness module. The module currently contains four different loudness models (two models are standardized).

5.2.1 Loudness prediction

To predict the loudness sensation for the selected signal, press the button “Compute” (see Figure 13). The result will be displayed as the overall loudness for both ears. For a mono signal there will be only one result, the overall loudness for both ears is divided by 2. The standard unit for the loudness sensation is *Sone*. It is possible to switch the loudness unit to *Phon* or to *Categorical Units* using the menu item “Preferences”. Apart from the overall loudness, the specific loudness (DIN 45631 and DIN 45631 / A1) and the dynamic loudness (DIN 45631 / A1 and Glasberg & Moore 2002) are calculated and displayed (see Figure 14).

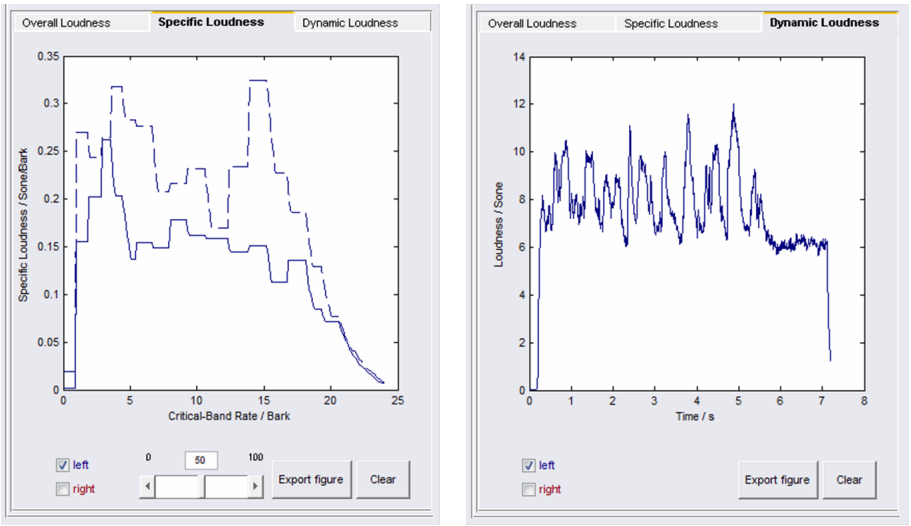

Figure 14: Zoom on the specific and dynamic loudness predictions from models appropriate for time-varying sounds.

Specific loudness

Within the implemented loudness models, the input signal is divided into psychoacoustically motivated frequency bands, like BARK [16] or the equivalent rectangular bandwidth (ERB) [17]. The loudness is calculated in each band resulting in the so-called specific loudness. To obtain the overall loudness for a given time sample, the specific loudness is added across bands. The SIP-Toolbox offers the possibility to display the specific loudness function (see Figure 14). You can display different time frames, divided in 1%-steps from the total signal length, using the slider above the specific loudness figure.

Dynamic loudness

The loudness-vs-time function is displayed in the panel “Dynamic Loudness”, see the right side of Figure 14 (only for loudness models for time-varying sounds).

5.3 Module Speech Quality

The module “Speech quality” contains several models and objective measures as shown in Figure 15. All models require a reference signal. You can select which signal (speech, noise or mixed signal) to be evaluated by comparison to the reference signal. All implemented speech quality model are described in the following.

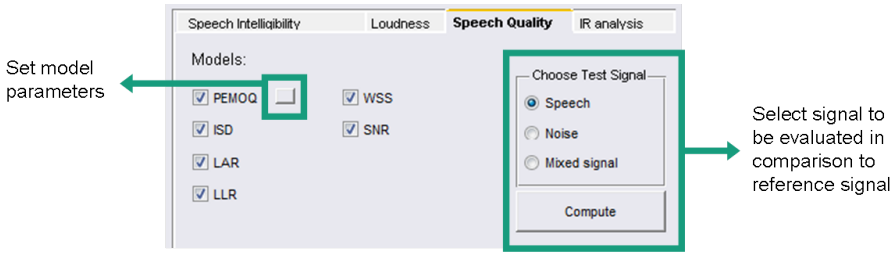

Figure 15: Zoom on the implemented models within the speech quality module.

PEMOQ_1: Model to predict speech quality based on the Oldenburger perception model [10, 11] using a linear cross-correlation. Within the model the option “SpeechQual” is used. You will find further information about PEMO-Q and the parameters in the PEMO-Q manual and in [13]. As result, a value of 1 indicates a perfect quality according to the reference signal (perfect match). The value 0 indicates no match with the reference.

PEMOQ_2: Model to predict speech quality based on the Oldenburger perception model [10, 11] using a linear cross-correlation. The difference to PEMOQ_1 is the usage of the option “AudioQual”. You will find further information about PEMO-Q and the parameters in the PEMO-Q manual and in [13]. As result, value of 1 indicates a perfect quality according to the reference signal (perfect match). The value 0 indicates no match with the reference.

LLR: Log Likelihood Ratio [12, p.40]. Comparison of two windowed speech signals (LPC-analysis) using the auto-correlation. Calculates the total correlation as a weighted sum of the correlations for each window width. The closer the LLR-results are to 0, the better is the quality rate.

ISD: Itakura-Saito Distance [12, p.50]. Prediction of quality (distance) based on a windowed LPC-analysis similar to “LRR”. The closer the ISD-results are to 0, the better is the quality rate.

LAR: Log Area Ratio [12, p.233]. Quality statement based on a windowed LPC-analysis of the signal and determination the distance of the linear prediction reflexion coefficients. The closer the LAR-results are to 0, the better is the match with the reference.

WSS: Weighted Spectral Slope Distance [12, p.56ff]. Separation of the test and the reference signal into 45 critical bands and determination of the intensities in every band. To predict a quality value the model determines the weighted distances of the spectral slopes in each frequency band. The closer the values are to 0, the better is the match with the reference.

SNR: Mean Segmental Signal-to-Noise-Ratio [12, p.45]. Determination of the segmental signal-to-noise ratio by calculating the SNR in overlapping parts of the signal, followed by averaging across segments.

5.3.1 Select the reference signal

The implemented models for the assessment of speech quality are based on a comparison between test signal and reference signal. Without a reference signal the prediction of the test signal quality is not possible. Therefore, if you want to proceed without a reference you will receive a warning. The selection of the reference signal is the same as for the test signal using a pop-up menu, illustrated in Figure 16.

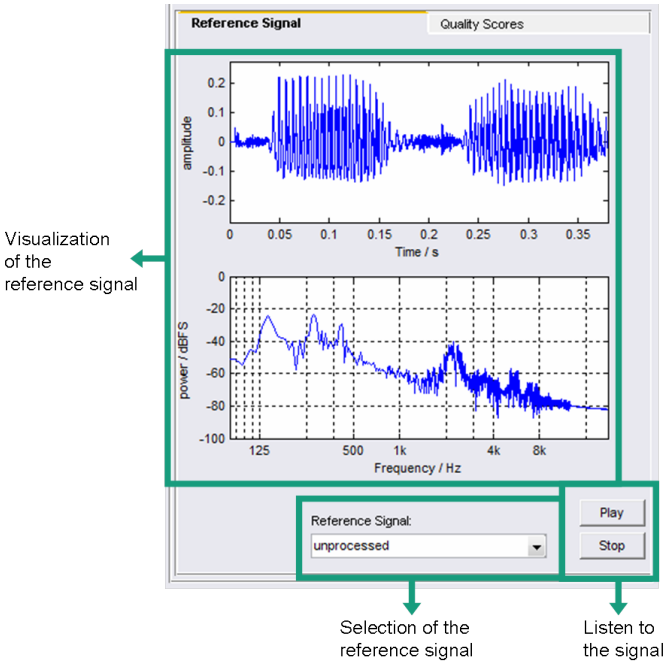

Figure 16: Selection of the reference signal for the speech quality module. The assessment of speech quality is based on a comparison between test and reference signal.

5.3.2 Assessment of speech quality

Like in all other modules of the SIP-Toolbox you can start the speech quality assessment by pressing the button “Compute”, see Figure 15. Make sure you have selected a reference signal. The results are displayed for each selected quality model and condition, as illustrated in Figure 17. Every calculation is set as a new condition and the results are displayed in the corresponding column. At maximum it is possibly to show 50 result values before it is necessary to save and/or clear the display. For a better presentation and to compare the results the SIP-Toolbox provides the possibility to plot all processed calculations in a separate figure by using the button “Plot”. Furthermore, it is possible to display all parameters from previous calculations, like signal names or SNR values, by moving the cursor over the text “cond” above the displayed results, see right side of Figure 17.

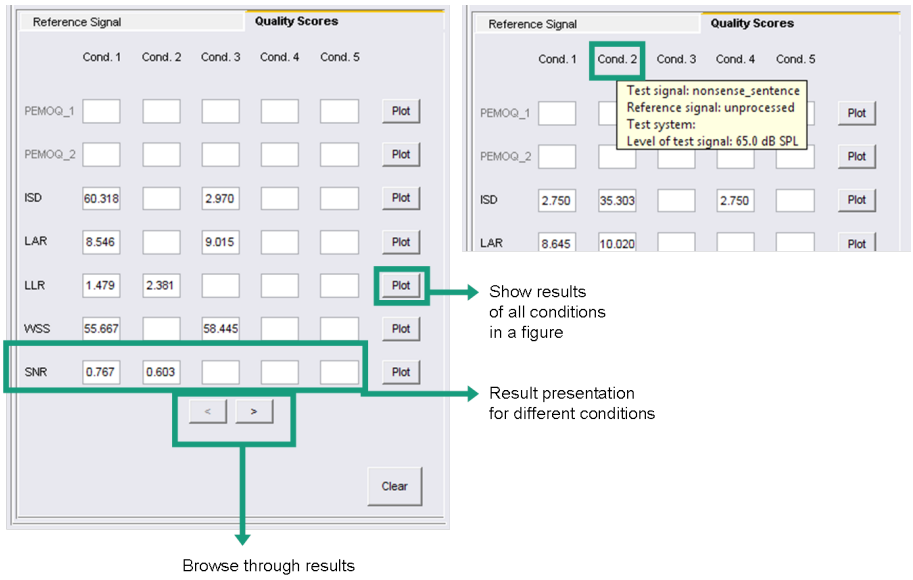

Figure 17: Detailed view on the result presentation of the speech quality module (left side). To display all set parameters used during the calculation, move the mouse cursor over the corresponding word "cond" (right side).

The SIP-Toolbox provides example files to test the speech quality module. The reference signal is called `unprocessed.wav` and the test signal is integrated by the name `processed.wav`. Both signals are located in the main folder or already included in the pop-up menu of the signal selection (demo version). For more detailed information about the speech quality models see the referenced literature.

5.4 Module Impulse Respone Analysis

The module "IR analysis" contains several objective measures for the room impulse response (RIR) analysis as shown in Figure 18. Some of these measures are useful to do a first quick evaluation of speech intelligibilty in the room represented by the RIR and others indicate if a room is suitable for musical presentation. The implemented RIR analysis techniques are described in the following.

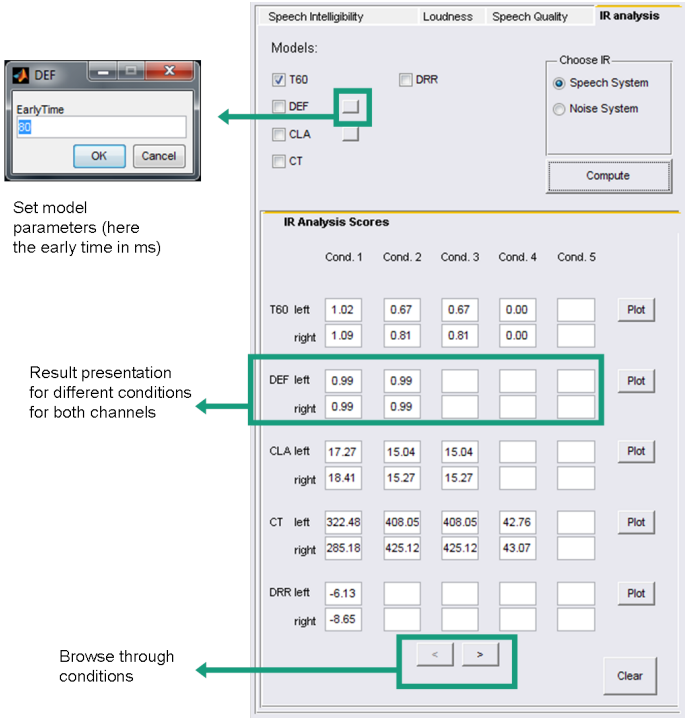

Figure 18: Zoom on the implemented impulse response analysis module. Within the module there are currently five different analysis functions available. For the models DEF and CLA, you can specify the early time in milliseconds as parameter.

T60: Reverberation time T_{60}. Reverberation is the persistence of sound in a particular space after the sound source is switched off. T_{60} is the time required for the energy of a sound to decay by 60 dB after being switched off. The algorithm calculates the time of a decay by 30 dB (5 to 35 dB) using the energy decay curve and interpolates the result to a decay by 60 dB. The unit of T_{60} is second as shown in Table 1, which contains some standard reverberation times (see also [15]).

purpose	T_{60} [s]
speech	0.8...1.4
symphonic music	1.1...2.6
organ music	1.3...3.5

Table 1: Desired reverberation times for different purposes. The times T_{60} depend on the volume and absorption properties of the room [15].

DEF: Definition (German: "Deutlichkeit") by THIELE, 1953. Generally, our hearing does not perceive reflections with delays of less than about 50 ms as separate acoustical events. Instead, such reflections enhance the apparent loudness of the direct sound, therefore they are often referred to as "useful reflections". The remaining reflections with longer delays are responsible for what is perceived as the reverberation of the room. The relative contribution of useful reflections may be characterized by several parameters derived from the impulse

response h(t). One of them is the "definition" or "Deutlichkeit" given by

$$\mathrm{DEF} = \frac{\int_0^{50\mathrm{ms}} \mathrm{h}(t)^2 \mathrm{d}t}{\int_0^{\infty} \mathrm{h}(t)^2 \mathrm{d}t} \cdot 100\% \; . \tag{5}$$

It can serve as an objective measure for speech intelligibility. The definition should be above DEF > 50 % for a good intelligibility.

CLA: Clarity (German: Deutlichkeitsmaß C_{50}; Klarheitsmaß C_{80}). The "clarity" is an objective measure which is used to characterize the transparency of speech (C_{50}) or musical presentations (C_{80}) as the energy ratio of the first 50 or 80 milliseconds to the overall energy of the room impulse response. It is defined by

$$\mathrm{CLA} = 10 \cdot \log_{10} \left[\frac{\int_0^{xx\ \mathrm{ms}} \mathrm{h}(t)^2 \mathrm{d}t}{\int_{xx\ \mathrm{ms}}^{\infty} \mathrm{h}(t)^2 \mathrm{d}t} \right] \mathrm{dB} \; , \tag{6}$$

where xx is the early time, normally 50 ms for the clarity of speech and 80 ms for the clarity of musical presentations. Desired values for a good clarity are $C_{50} > 0$ dB for speech and $C_{80} \approx 0$ dB ±4 dB for music.

CT: Center time (German: Schwerpunktszeit). The "center time" is an objective measure used to characterize the room for the purpose of musical presentations. It is defined by

$$\mathrm{CT} = \frac{\int_0^{\infty} t \cdot \mathrm{h}(t)^2 \mathrm{d}t}{\int_0^{\infty} \mathrm{h}(t)^2 \mathrm{d}t} \; . \tag{7}$$

An appropiate value for the center time is CT ≈ 120 ms ±30 ms.

DRR: Direct-to-Reverberation Ratio. This objective measure discribes the energy ratio between the intensities of the direct sound and reverberation in dB. It is known to be an important acoustic cue for sound source distance perception.

5.4.1 Perform room impulse response analysis

Like in all other modules of the SIP-Toolbox you can start the impulse response analysis by pressing the button "Compute" (see Figure 18). The results are displayed for each objective measure and condition. Every calculation is set as a new condition and the results are displayed in the corresponding column. At maximum it is possibly to show 50 result values before it is necessary to save and/or clear the result display. For a better presentation and to compare the results the SIP-Toolbox provides the possibility to plot all processed calculation in a separate figure by using the button "Plot". Furthermore, it is possible to display all parameters of previous calculations, like signal names or SNR values, by moving the cursor over the text "cond" above the displayed results.

6 Main menu bar

The main menu bar offers an easy and quick access to the global function within the SIP-Toolbox. All menu items are described in the followings.

6.1 Menu item Setup

6.1.1 Load

The menu item "Load" offers the possibility to load and display saved results of previous predictions. Additionally, it is possible to load custom signal lists (see next section).

6.1.2 Save

The menu item "Save" offers the possibility to save all kinds of data generated with the SIP-Toolbox.

Data: Save results of the different modules and objective measures.

Signals: Save signal (speech, noise or mixed signal) as a dual-channel wave file according to the displayed time structure.

Custom information: It is possibly to save a custom signal list. If this list is loaded in the SIP-Toolbox again, the custom signals will be listed and integrated in the signal selection. This menu item also allows you to save "information window" entries as text file.

6.1.3 Batch Processing

Batch processing speech intelligibility

The batch processing within the SIP-Toolbox is based on a control text file. For every module there is an example text file (for speech intelligibility it is called `testSI_batch_in.txt`). Here you define parameters for the batch processing. Figure 19 shows the structure of this text file to perform a speech intelligibility batch processing.

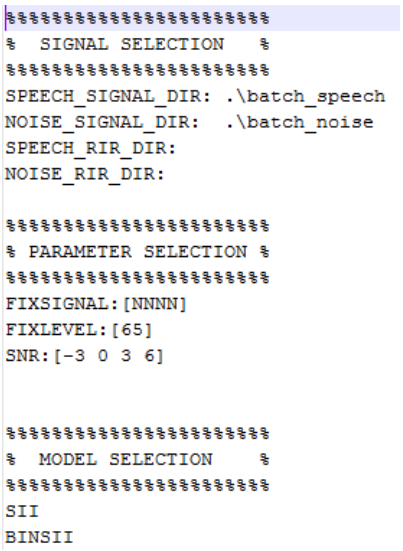

Figure 19: Example file (`testSI_batch_in.txt`) for the speech intelligibility batch processing. (`N`: Noise, `S`: Speech)

The file structure includes three parts:

`SIGNAL SELECTION:` Here you have to enter the paths of the signals (speech and noise and optional impulse responses). You can use absolute or relative paths. To set a relative path please use following syntax `.\HOME\...`. The number of signals in each folder must either be 1 or N. For example, if you use eight different speech signals (`01_speech`, `02_speech`, ..., `08_speech`) and eight different noise signals (`01_noise`, `02_noise`, ..., `08_noise`), then the selected model predictions are calculated for the eight pairs. It is therefore important that you name the files consistently, for example by numbering them. If you place only one speech signal in the speech folder, this signal is used with all eight noise signals.

`PARAMETER SELECTION:` To evaluate the speech intelligibility, you have to decide which signals are fixed at a defined level. Similarly you have to define a signal-to-noise ratio for every signal combination. As for the signals, the number of parameters has to be equal to the number of signals. If you want to use just one parameter value for all signal pairs, it is only necessary to set one value.

`MODEL SELECTION:` Define here the objective measures and models you want to work with.

To start the batch processing select the item "Setup" → "Batch Processing" → "Speech intelligibility" from the menu bar. Now there is a dialog to select the batch processing text file. Select the text file, for instance `testSI_batch_in.txt`, and the assessment of the speech signal in noise will start. You will be informed about the completion on the screen. The results will be also saved in a text file, labeled like the start file with the name extension `*_out.txt`. Every batch processing is separated by a date and time stamp. Figure 20 shows an example of a result file of a speech intelligibility batch processing.

```
% BATCH MODE SpeechIntel
% Results computed from: testSI_batch_in.txt
% Time stamp: 24-Mar-2010  14:40:51:

% AI:
0.077, 0.077, 0.171, 0.271  % left channel
0.077, 0.077, 0.146, 0.271  % right channel

% SII:
0.155, 0.155, 0.335, 0.440  % left channel
0.155, 0.155, 0.305, 0.440  % right channel

% STI:
0.242, 0.242, 0.412, 0.503  % left channel
0.242, 0.242, 0.385, 0.503  % right channel

% BATCH MODE SpeechIntel
% Results computed from: testSI_batch_in.txt
% Time stamp: 24-Mar-2010  14:41:45:

% AI:
0.077, 0.077, 0.171, 0.271  % left channel
0.077, 0.077, 0.146, 0.271  % right channel

% SII:
0.155, 0.155, 0.335, 0.440  % left channel
0.155, 0.155, 0.305, 0.440  % right channel

% STI:
0.242, 0.242, 0.412, 0.503  % left channel
0.242, 0.242, 0.385, 0.503  % right channel
```

Figure 20: Result file (`testSI_batch_in_out.txt`) of a speech intelligibility batch processing.

Batch processing speech quality

An example control file of the speech quality batch processing is shown in Figure 21.

```
%%%%%%%%%%%%%%%%%%%%%%
%  SIGNAL SELECTION   %
%%%%%%%%%%%%%%%%%%%%%%
TEST_SIGNAL_DIR: .\batchSQ_test
REFERENCE_SIGNAL_DIR: .\batchSQ_ref
IR_DIR:

%%%%%%%%%%%%%%%%%%%%%%
% PARAMETER SELECTION %
%%%%%%%%%%%%%%%%%%%%%%
LEVEL:[65]

%%%%%%%%%%%%%%%%%%%%%%
%  MODEL SELECTION    %
%%%%%%%%%%%%%%%%%%%%%%
PEMOQ:{'MaxLag','150'}
ISD
```

Figure 21: Example file (`testSI_batch_in.txt`) for speech quality batch processing.

For speech quality batch processing, the only parameters to be defined are the level of the test signal and the objective measures you want to calculate. There are optional parameters for the assessment of speech quality with PEMO-Q, which have to be defined in curled brackets as shown in Figure 21. For more information about these parameters see the PEMO-Q manual.

Batch Processing Hagermann & Olofsson

The batch processing according to the method of Hagermann & Olofsson is explained in detail in section 4.

6.1.4 Hagermann & Olofsson

The menu functions for the method of Hagermann & Olofsson are explained in section 4.

6.1.5 Preferences

In the "Preferences" menu, you can specify general parameters for your usage of the SIP-Toolbox, which will be kept after closing the program. Here you can select which spectrum analysis to be used by the SIP-Toolbox and set parameters like "FFT-Size" and "Window type". You can also set parameters used in the module "Loudness" like leading and tailing zeros for the selected signals or specify the loudness unit.

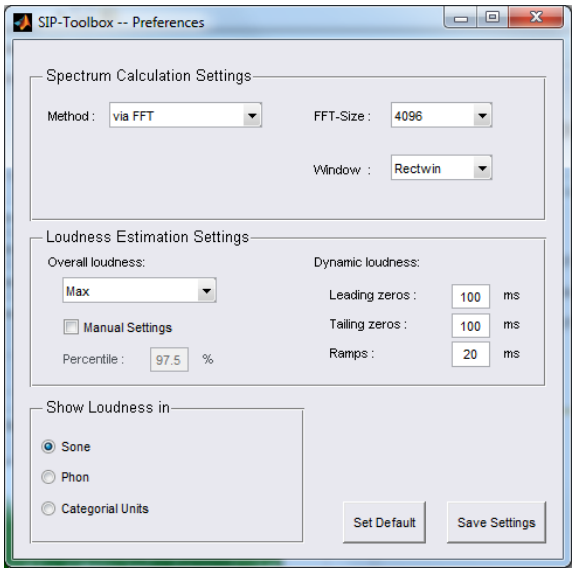

Figure 22: Preference menu of the SIP-Toolbox with possibilities to define parameters used by the signal visualization and the loudness estimation.

6.1.6 Reset

With the menu item "Reset" the software will be restarted with default values. Saved settings will be kept.

6.2 Menu item Audiogram

Set Audiogram

After calling the main menu item "Set Audiogram" a new window is opened as shown in the left side of Figure 23. This audiogram visualization allows you to manually set any hearing loss for the standard frequencies of a pure-tone audiogram via mouse input.
Additionally, it is possible to create a tone audiogram according to ISO1999:1990. Another window is opened, as shown in the right side of Figure 23. Using the age, noise exposure time in years

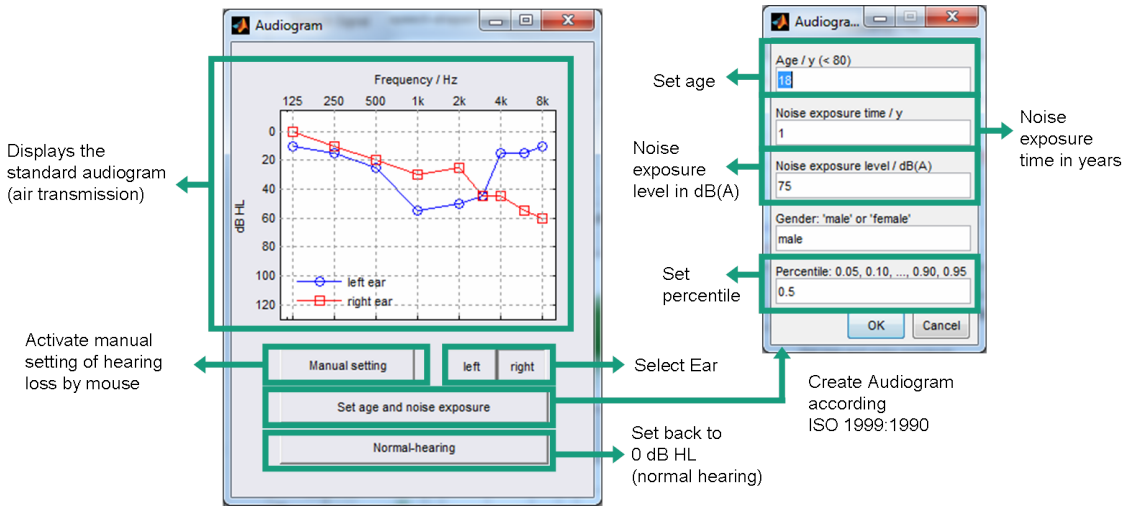

Figure 23: The SIP-Toolbox offers the opportunity to set an individual hearing loss according to the method ISO1999:1990. It is also possible to manually set any hearing loss via mouse input.

and the average expositions level, a statistically expected hearing loss is calculated and displayed in the audiogram.
After having set the audiogram, the new values are automatically included in any subsequent model predictions for models which include the hearing threshold (e.g. the SII).

6.3 Menu item Binaural Options

Set Binaural Configuration

After calling the main menu item “Set Binaural Configuration” a new window is opened, as shown in Figure 24. This window offers a visual way to configure different acoustical situations. The graphical binaural configuration is directly linked to the transfer function selection of the SIP-Toolbox main window and will only work with the provided transfer functions “anechoic” “cafeteria” and “office”. The binaural configurations can be saved and loaded again into the software.

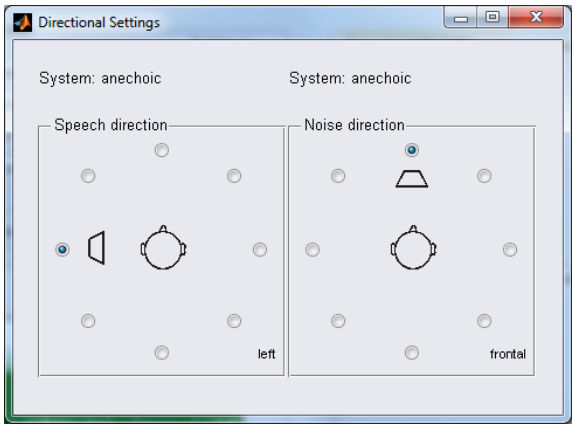

Figure 24: Graphical binaural configuration for the provided transfer functions “anechoic”, “cafeteria” and “office” within the SIP-Toolbox.

References

[1] Hagermann, B. and Olofsson, A (2004). “A method to measure the effects of noise reduction algorithms using simultaneous speech and noise”, Acta Acoustica United With Acustica, Vol.90, 356-361

[2] ANSI S3.5-1969 (1969). “Methods for the calculation of the articulation index”, American National Standards Institute

[3] ANSI S3.5-1997 (1997). “Methods for calculation of the speech intelligibility index”, American National Standards Institute

[4] IEC 60268-16 (2003). “Objective rating of speech intelligibility by speech transmission index”, International Electrotechnical Commission, 3, rue de Varembé, PO Box 131, CH-1211 Geneva 20, Switzerland

[5] Beutelmann, R. and Brand, T. (2006). “Prediction of speech intelligibility in spatial noise and reverberation for normal-hearing and hearing-impaired listeners,” J. Acoust. Soc. Am 120(1), 331-342.

[6] DIN 45631 / A1: 2010. “Calculation of loudness level and loudness from the sound spectrum - Zwicker method - Amendment 1: Calculation of the loudness of time-variant sound´´. Deutsches Institut für Normung

[7] DIN 45631 / ISO 532B: 1991-03. “Calculation of loudness level and loudness from the sound spectrum - Zwicker method.´´. Deutsches Institut für Normung

[8] Moore, B.C.J. and Glasberg, B.R. (2002). “A model of loudness applicable to time-varying soands”, Journal Audio Engineering Society, 50(5):331 - 334.

[9] Moore, B.C.J., Glasberg, B.R. and Baer, T. (1997). “A model for the prediction of threshold, loudness and partial loudness”, Journal Audio Engineering Society, 45(4):224 - 240.

[10] Dau, T., Püschel, D. and Kohlrausch, A. (1996). “A quantitative model of the ’effective’ signal processing in the auditory system. I. Model structure,” J. Acoust. Soc. Am 99(6), 3615-3622

[11] Hansen, M. and Kollmeier, B. (2000). “Objective modelling of speech quality with a psychacoustically validated auditory model,” J. Audio Eng. Soc., vol. 48(5), 395-409

[12] Quackenbush, S. R., Barnwell, T. P. and Clements, M.A. (1988). “Objective Measures of Speech Quality”. Prentice Hall Advanced Reference Series, Englewood Cliffs, NJ, ISBN: 0-13-629056-6.

[13] Huber, R. (2003). “Objective assessment of audio quality using an auditory processing model,” PhD thesis, Universität Oldenburg

[14] Hansen, M. (1998). “Assessment and prediction of speech transmission quality with an auditory processing model”, PhD thesis, Universität Oldenburg

[15] DIN 18041:2004-05 (2004). “Acoustic quality in small and medium-sized rooms”. Deutsches Institut für Normung

[16] Zwicker, E. (1961). “Subdivision of the audible frequency range into critical bands”, J. Acoust. Soc. Am. 33

[17] Moore, B.C.J. and Glasberg, B.R. (1983). “Suggested formulae for calculating auditory-filter bandwidths and excitation patterns”, J. Acoust. Soc. Am. 74: 750-753